WRITE-ONCE MEMORY DEVICE INCLUDING NON-VOLATILE MEMORY FOR TEMPORARY STORAGE

BACKGROUND

[0001] Portable devices such as PDAs, handheld computers, digital cameras and digital music players include memory for storing data, digital images and MP3 files. Different types of memory are available for these portable devices. Conventional memory types include flash memory, mini-hard drives, mini-compact discs, and magnetic tape. However, each of these memory types has one or more of the following limitations: large physical size, low storage capacity, relatively high cost, poor robustness, slow access time and high power consumption.

[0002] Solid state diode-based one-time programmable (OTP) memory is disclosed in assignee's U.S. Serial No. 09/875,356 filed June 5, 2001. Compared to the conventional memory, the diode-based memory has a high shock tolerance, low power consumption, fast access time, moderate transfer rate, and good storage capacity. The diode-based memory can fit into a standard portable interface (e.g., PCMCIA, CF) of a portable device.

[0003] User data is written to the OTP memory in blocks. Before the blocks are written to the OTP memory, however, error correction data such as ECC code words are added. Errors can occur due to defects in the storage medium and noise in the read channels. If errors occur in the user data, the error correction data allows the errors to be corrected.

[0004] To calculate the error correction data, a full block is needed. A typical block size is 32 kilobytes.

[0005] The block size is always the same. However, data file sizes of less than one kilobyte are not uncommon. If only a few bytes of user data are to be written to the OTP memory, padding (i.e., extra bytes) may be added to the user data to reach the 32 KByte block size. Once full block size has been reached, the error correction data is generated.

[0006] Memory is wasted if the user data is padded. A more efficient approach is needed, especially where large numbers of small files are involved.

SUMMARY

[0007] According to one aspect of the present invention, a data storage device includes write-once memory; non-volatile memory; and a circuit for writing user data to the write-once memory and at least one of the user data and error correction data to the non-volatile memory. Other aspects and advantages of the present invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating by way of example the principles of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] Figure 1 is an illustration of a multi-level solid state OTP memory device.

[0009] Figure 2 is an illustration of memory and circuitry formed in a substrate of the memory device.

[0010] Figure 3 is an illustration of a first method of writing a block of data to the memory device.

[0011] Figure 4 is an illustration of an RS-PC block.

[0012] Figures 5a and 5b are illustrations of a second method of writing a block of data to the memory device.

DETAILED DESCRIPTION

[0013] As shown in the drawings for the purpose of illustration, the present invention is embodied in a solid state OTP memory device. The memory device can store small files to write-once memory, while preserving error correction capability, but without adding significant padding. Consequently, write-once memory is used more efficiently, especially where large numbers of small files are involved. Moreover, conventional error correction algorithms may be used.

[0014] Reference is made to Figures 1 and 2. A multi-level solid state memory device 110 includes a stack of levels 112 of write-once memory. The stack is formed on a substrate 114. Each level 112 includes main memory and address logic. Details of an exemplary level 112 are disclosed in assignee's U.S. Serial No. 09/911,974 filed July 24, 2001 and incorporated herein by reference.

[0015] Control/interface circuitry 210 and a data buffer 212 are formed in the substrate 114. The control/interface circuitry 210 performs functions for operating the device 110. These functions include setting write voltages, setting write enable lines and controlling power sense striping, memory addressing by converting logical addresses to address line patterns required to access physical memory locations, and data read processing of sense line outputs.

[0016] The levels 112 are connected to the control/interface circuitry 210 by a memory system interface 116. The system interface 116 includes address lines. Address logic on each level 112 shares the same address lines. When a word is written to the write-once memory, the bits of the word are not stored in contiguous memory elements. Instead, the bits are stored on different levels 112.

[0017] The control/interface circuitry 210 assembles user data into user data blocks (in the data buffer 212), generates error correction data (e.g., error code correction or "ECC" code words), and writes the user data and error correction data to the write-once memory as described below.

[0018] The substrate 114 also contains a small amount of non-volatile memory 214 (e.g., flash memory) for storing at least one of user data and the error correction data. The size (i.e., storage capacity) of the non-volatile memory 214 will depend upon how incomplete blocks of user data are handled, as will be described below. Typically, storage capacity of the non-volatile memory 214 will not exceed the size of a user data block (e.g., 32 KBytes).

[0019] Additional reference is made to Figure 3, which illustrates a first method, performed by the control/interface circuitry 210, for writing user data to the write-once memory. The control/interface circuitry 210 receives user data (e.g., from a host) (310) and buffers the user data in the buffer 212 (312). The control interface circuitry 210 determines whether the buffered user data forms a full block (314). If the buffered user data forms a full block, the control/interface circuitry 210 generates error correction data (316), and writes the user data and the error correction data to the write-once memory (318).

[0020] If the block of user data is incomplete (that is, not enough user data is present to make a full block) (314), the control/interface circuitry 210 writes the user data to the non-volatile memory 214 (320).

[0021] When the control/interface circuitry 210 receives new user data, it moves the user data from the non-volatile memory 214 to the buffer 212, and adds the new user data until a full block is formed (324). Once a full block is formed, error correction data is generated for the full block (316), and the user data and the error correction data are written to the write-once memory (318).

[0022] A variety of encoding methods are available for generating the error correction data. For example, Reed-Solomon Product Code ("RS-PC") encoding may be used. An RS-PC block and an alternative method for writing to an OTP device will now be described. The alternative method will be described in connection with RS-PC encoding.

[0023] Reference is now made to Figure 4, which shows an RS-PC block 410. A total of sixteen 2KByte sectors are arranged into a 32KByte block of user data 412. RS-PC encoding is performed on each 32KByte block of user data. The RS-PC encoding is two-dimensional. An RS-PC code word for a row may be computed after the row is filled, and the code word is appended to the end of that row. The last column 414 of the RS-PC block 410 contains these "first level" code words. After all rows of user data (and their code words) have been filled, RS-PC code words for the columns are generated, and appended to the end of the columns. The last row 416 of the RS-PC block 410 includes these "second level" code words.

[0024] Reference is now made to Figures 5a and 5b, which show an alternative write method performed by the interface/control circuitry 210. The interface/control circuitry 210 receives user data (510) and writes the user data to the buffer 212 (512). If the buffered user data forms a full block (514), the control/interface circuitry 210 performs RS-PC encoding on the full block (516), and writes the resulting RS-PC block to the write-once memory (518).

[0025] If the buffered user data does not form a full block (514), padding (e.g., 0's) is added to the buffered user data to form a full block (520), and RS-PC encoding is performed on the padded block (522). The buffered user data and final code words are written to the write-once memory (524). The final code words in this example would be those first level RS-PC code words at the end of complete lines of user data.

[0026] The data just written to the write-once memory is stored between starting and ending addresses in the write-once memory. Those starting and ending addresses are written to the non-volatile memory 214 (526). Further, temporary RS-PC code words are written to the non-volatile memory 214 (526). The temporary code words in this example would be all second level RS-PC code words and those first level RS-PC code words at the end of incomplete lines of user data.

[0027] When new user data is ready to be received (block 530), the starting and ending addresses and the temporary RS-PC code words are accessed from the non-volatile memory 214 (532), user data and final code words between the starting and ending addresses is accessed from the write-once memory (534), the same padding is added to the accessed user data to form a full block (536), and the full block is error corrected using the accessed RS-PC code words (538). The error-corrected user data (but not the padding or the code words) is stored in the data buffer 212 (540).

[0028] The new data is received and overwrites the padding (542), and RS-PC encoding is performed (544). If all of the padding has been overwritten (546), the new final code words and the new user data are written to the write-once memory (548).

[0029] If all of the padding has not been overwritten (546), the user data that overwrote the padding and the new final code words are stored in write-once memory (550). In addition, new temporary code words and the new end address are written to the non-volatile memory 214 (552).

[0030] The method of Figures 5a and 5b is not limited to RS-PC blocks. It may be used in connection with any error correction scheme.

[0031] The method of Figures 5a and 5b may be modified as follows. As one example, all code words (temporary and final) are written to the non-volatile memory.

[0032] As another example, only complete rows of user data and their corresponding final code words are written to the write-only memory. Incomplete rows of user data are written to the non-volatile memory 214, along with the starting/ending addresses and the temporary code words. Before additional data is received, the user data and the final code words are retrieved from the write-once memory, the incomplete row and temporary code words are retrieved from the non-volatile memory, the padding is added, and error correction is performed.

[0033] Although the memory device was described in connection with solid state OTP memory, it is not so limited. The memory device may be any data storage device that includes write-once memory, a small amount of non-volatile memory, and a circuit for writing user data to the write-once memory and storing in the non-volatile memory at least one of user data and error correction data.

[0034] The present invention is not limited to the specific embodiments described and illustrated above. Instead, the present invention is construed according to the claims that follow.